LANES 3 User Manual

Network Simulation Group
Sterling Software
1121 San Antonio Road
Palo Alto, CA 94303
(415) 964-9900

Prepared for
Ames Research Center
Contract NAS2-11555

November 1986

Contents

List of Figures

Chapter 1

The Purpose of LANES 3

1.1 Introduction

NASA's commitment to Space Station requires the development of sophisticated data processing systems beyond the range of technology which is currently available. Computer complexes aboard Space Station will use high speed local area networks (LAN) transmitting between stations. The appropriate technology is now being developed. LANES 3 provides a method for simulating anticipated network conditions.

LANES 3 is a model of a Fiber Distributed Data Interface (FDDI) ring network configuration, and of a Star*Bus.[1] It incorporates the link layers together with a common network and load layer. It augments the features of LANES 1 and LANES 2.

Both LANs are designed to transmit at very high speeds using fiber optic cable. The FDDI model is based on the American National Standard FDDI Token Ring Media Access Control X3T9/84-X3T9.5/83-16 Rev. 10, February 28, 1986 (ANSI specification). The Star*Bus model is based on a prototype hardware version of the protocol developed by the Sperry Corporation, under contract to NASA, Goddard Space Flight Center.

The simulation is designed to determine performance characteristics of FDDI and Star*Bus under a variety of loading conditions. It allows the user to objectively compare the performance of FDDI and Star*Bus.

1.2 Introduction to the LANES 3 Network model

The LANES 3 model is presented in menu-driven screens which allow the user to enter or retrieve input and output parameters and to run simulations.

Output results are stored using the relational database product, INGRES.[2] The

[1] Star*Bus is also referred to as Fiber Optic Demonstration Systems (FODS)in this document and in the network model.

[2] INGRES is a product of Relational Technology, Incorporated.

user interface provides a set of basic instructions and allows the user to see simulation results within minutes. The process is:

1. enter parameters for simulation;
2. run simulator;
3. analyze results.

Inputs characterize the configuration of the network, define the load, and control the simulation. Network parameters include: the number of stations, the transmission rate, and the size of buffers, amoung others. The load on the network is specified by individual stations. The station specific parameters include a choice among probability distributions, and specific loading between stations. The control parameters enable simulations to be run for varying amounts of time and reporting intervals.

Results from the simulation can be obtained in printed reports, reviewed directly on the LANES 3 screens, reviewed in a text editor or manipulated directly in the database.

Project members are: Terry Grant (Ames-RII), Marjory Johnson (RIACS), Keith Richardson (Sterling Software), and Sharon Doubek (RECOM Software, Inc.). Larry Webster (Ames-RII) and Joe Jordan (Sterling Software) previously contributed to the LANES projects.

This documentation covers the basic issues which will be of interest to users. Further questions can be addressed to Terry Grant at NASA/Ames Research Center, telephone: (415) 694-6525.

1.3 User Manual Layout

A term that is used in the LANES 3 simulation that has additional connotations beyond general usage, is CAPITALIZED when it first appears in the text. A technical term is not capitalized when its meaning is within general usage.

The users manual outline of the LANES 3 architecture is followed by a detailed explanation of the system and the user inputs. If the user with access to the LANES 3 account would like to run the LANES 3 simulation immediately using default values, they should turn to chapter 5 "Running a Demo". Chapter 5 should be read carefully before attempting to run LANES 3. Running a demonstration would provide a quick way to understand the framework of the simulation. Descriptions of the parameters are located in the Glossary (page 61).

The explanation of the LANES 3 architecture progresses from the highest level (Load Layer) down to the lowest level (Physical Layer). For each layer both transmitting and receiving functions are discussed at the same time.

Chapter 2

Architecture and the LANES 3 Model

Our sources for details of the Star*Bus Media Access Control protocol are discussions with the Sperry Corporation. The Sperry Corporation has developed and implemented the Star*Bus protocol under contract to Space Systems Division of NASA Goddard Space Flight Center.

Our sources for details of the FDDI Media Access Control protocol are "FDDI Token Ring Media Access Control," Draft Proposed American National Standard, X3T9.5/83-16, Rev. 10, February 28, 1986, and conversations with various members of the ANSI X3T9.5 committee which is developing FDDI. LANES 3 provides some flexibility in specifying protocol parameters and details.[1] The method chosen for modeling the functions of Load and Network Layers, and some of the Link Layer functions, results from discussions within the Data Network Group at NASA/Ames Research Center.

2.1 Outline of LANES 3 Architecture

The LANES 3 model simulates the LOAD, NETWORK, LINK, and PHYSICAL layers systems architecture. This architecture takes MESSAGES from the host computers at stations on the network (modeled as load) to be transmitted over the network. These messages are broken down into frames, which are stored in the Network Layer and the Link Layer buffers until they are transmitted. At the destination station, frames are reassembled into messages at the Network Layer and experience ABSORPTION delay before they are passed to the Load Layer.

The load is determined by user input parameters. A distinction is made between messages which are of type SYNCHRONOUS or ASYNCHRONOUS. Synchronous Messages have a higher priority and are released from the Network Layer to the Link Layer before Asynchronous Messages. Two PRIORITY LEVELS within Asynchronous

[1] Transmission rate and header sise are amoung Network Wide input parameters that provide flexibility.

Message types may also be specified. In Star*Bus frames are not prioritized at the Link Layer, but in FDDI the synchronous/asynchronous distinction is maintained. The Link Layer queue is modeled as a first-in-first-out queue. A frame is transmitted according to specifications established by the FDDI or Star*Bus protocol.

Once the network characteristics have been specified, the simulation is run without interruption. A model is created of the specified characteristics using random number generators and mathematical functions to determine message arrival in the network, the message type, message size and messages propagate according to the appropriate protocols and hardware functions.

Output is available both as a summary of results, portraying maximum, minimum, average, and standard deviation values, or as a TRACE which records each action taken by the simulation as it happens. The summary results are available for further analysis in the database. The Trace is a text file which the user can review in detail using a text editor after the simulation has finished.

The summary results from a simulation can be loaded into the network model database for processing by the Analysis screens and reports, or report information can be read directly from the screen at the conclusion of a simulation. In the Analysis portion of the network model, these summary results can be permanently stored in the database for direct comparison to other simulation runs.

The next chapter of the LANES 3 manual includes a detailed description of the network.

Chapter 3

Details of Implementation

3.1 Description of Implementation

LANES 3 simulates the physical, link, and network layers of the ISO-OSI paradigm, with the upper four layers combined into a single entity, the Load Layer.

Input parameters are either STATION SPECIFIC, applying only to a single station, or NETWORK WIDE, applying to the entire network.

3.2 Load Layer

In the simulated communications load on the network, each message is characterized by type, (Synchronous, Asynchronous with priority 2 or Asynchronous with priority 1), size, source and destination stations. Each type of message is characterized by statistical distributions on the interarrival times, message sizes, and destinations. Other statistical distributions for each station determine the absorption times (see definition below) for each type of message which reaches it.[1]

Messages coming from the host are of sizes specified by the user in the Station Specific parameter MESSAGE SIZE. The parameter HEADER SIZE is a Network Wide parameter. There is an infinite amount of space at the Load Layer of the sending station to buffer messages. At the receiving station there is a delay of ABSORPTION TIME for the processing of each message before acceptance into the Load Layer. Acceptance into the network layer depends on the availability of the (finite) space in the network layer transmit and receive buffers.

INTERARRIVAL TIME is the amount of time between the end of generation of a message and the end of the generation of the next message of the same type at the same

[1] The statistical distributions for the following Station Specific parameters are each given independent random number seeds: Synchronous Interarrival, Asynchronous Type 2 Interarrival, Asynchronous Type 1 Interarrival, Message Sise, and Absorption Time. This way a change to one series will affect only that station and its parameter.

station. A message is generated at a station's Load Layer and is queued immediately in the Network Layer receive buffer, provided there is space for the entire message.

In the simulation a message is sent by a transmitting station and destined for one other station. The sender cannot transmit messages to itself.[2]

The following distributions are available for Synchronous and Asynchronous Message Interarrival Time, Absorption Time, and Message Size:

1. Constant.
2. Exponential.
3. Normal (Gaussian).
4. Uniform.

Stations can be divided into up to ten classes. Messages generated by a station are uniformly sent to all stations within the specified destination class. The simulation defaults to one class.

3.3 Network Layer

The Network Layer does not model a particular protocol but instead it implements buffering concepts of interest to NASA/Ames Research Center's study project. Two separate buffers for the Network Layer exist at each station, one buffer to hold messages being transmitted[3] and the other buffer is to hold messages that are received. The capacities are Station Specific input parameters.

The user controls the size of the frame with the Network Wide parameters, Largest Frame Size, and Header Size. The Station Specific parameters Asynchronous Message Size and Synchronous Message Size control the size of the messages at the Load Layer.

3.3.1 Message Buffer Allocation

As a message arrives from the load buffer at the Network Layer, in the transmitting station, space for the entire message is reserved. If there is not enough space available at the Network Layer, the message does not leave the load buffer.

The space occupied once the message has been accepted is held until duplicates of all the frames for that message have been sent, and an acknowledgment has been made by the receiving station. Only at this point are the master copies of the frames deleted at the sending station and the space freed.

[2] In both the FDDI and Star*Bus simulations all other stations receive a transmitted message. In Star*Bus the frame is ignored by all stations except the receiving station. In FDDI the frame is passed completely around the ring.

[3] Messages can be subdivided into frames for transmission as needed by the Link Layer protocol.

At the receiving end when the Network Layer accepts the first frame of a message the space needed for storage of the entire message is allocated. If there is not enough room for the entire message the frame is rejected.

The user may choose to implement a BACKOFF period to start when a transmitter receives a frame rejection flag. The backoff affects only the station which experienced the frame refection. The length of the Backoff Period is a Network Wide input parameter. During this time frames which are destined for the receiving station may not be released from the network transmission buffers. The Backoff does not affect frames already sent.

The Backoff period expires under two situations:

1. at the end of the Backoff time;

2. when a frame unaffected by the Backoff Timer is accepted by the receiver, and the transmitter gets back the Acknowledgment.

The function of the Network Layer queue during the Backoff period is unchanged except for the frozen frames affected by the Backoff. If the user does not wish to make use of this feature the Backoff period can be set to zero.

3.4 Link Layer

A frame is the unit of data transmitted over the channel. It consists of a header and a trailer, whose size is specified by the Network Wide input parameter HEADERSIZE, and an information field, whose size is specified by the Network Wide input parameter LARGESTFRAME. Specific transmission rules are dependent on the particular protocol being modeled.

3.4.1 Star*Bus Link Layer

Star*Bus is a Carrier-Sense Multiple-Access with Collsion Detection and contention resolution via Time Slots (CSMA/CD/TS) access protocol. There are two modes, random access mode and controlled access mode. In random access mode each station continually monitors the channel. A station with data may transmit if the channel has been idle for at least gap time, which is a user input.[4]

When a collision occurs, the resulting noise burst propagates to every station on the network. As each of the contending senders detects this, it immediately stops transmission and broadcasts a jamming pulse. Then each station is assigned a unique timeslot,[5] during which it can access the channel without contention. Each station starts an internal counter, which increments at the end of each successive timeslot.

[4] The input parameter GAPTIME can be found on the Star*Bus Network Wide input screen.

[5] Star*Bus Network Wide input parameter TIMESLOTWIDTH measured in microseconds.

When its count reaches a particular station's timeslot index, the station is allowed to transmit. During a transmission the countdown through the timeslots is suspended. At the end of the transmission, each station waits a gap time before resuming the countdown with the next station's timeslot.

At the end of the last timeslot, when the local count has gone over the maximum allowable number of stations on the network (32), each station waits a random delay before switching back into random access mode. Each random delay consists of a random number of periods of length RDMSTEPSIZE, a Network Wide input parameter. The random number is drawn from a uniform distribution on the range of integers from 1 to an input upper limit which is a Star*Bus Network Wide input parameter MAXTIMESTEP. If a station detects channel traffic while waiting its random delay, it immediately switches into random access mode, truncating the random delay.

At the Link Layer in Star*Bus there is a transmission buffer which can hold a single frame. When a frame transmission ends, that frame is removed from the Link Layer buffer and another frame enters from the Network Layer transmit buffer. A Network Wide input parameter called BUFFERDELAY is the time required to load a frame from the Network Layer to the Link Layer.

All stations receive a frame that is transmitted. Only the destination station copies the frame into its Network Layer receive buffer. An ACK or NACK is sent after a dely of 0.5 microseconds. The ACK/NACK is the same size as the user specified parameter "headersize".

3.4.2 FDDI Link Layer

FDDI is a timed token protocol. Station timers regulate the amount of time that each individual station can transmit, based on the observed network load. The TARGET TOKEN ROTATION TIME (TTRT), which is the expected token rotation time, is negotiated by the stations during ring initialization according to the FDDI protocol; it is a user input for the simulation. [6] Each station has a TOKEN ROTATION TIMER (TRT), a Synchronous Bandwidth Allocation, and a Late Count flag, which regulates the station's access to the channel. TRT is initialized to 0, counts upward, and expires when it reaches TTRT. The token is considered to arrive at a station on time if its TRT has not expired since it last received the token; otherwise, it is considered to be late. The Late Count Flag is set to 1 (one) if the token arrives late and is set to 0 (zero) if the token arrives on time.

A user parameter PERCENTSYN[7] specifies the percentage of TTRT that can be assigned to the various stations for transmisssion of highest priority (i. e. synchronous) frames. Each station is assigned a share of this synchronous bandwidth for its individual use. A station may use up to its Synchronous Bandwidth Allocation to transmit synchronous frames each time it receives the token.

---

[6] The data entry field TOKNROTATION is located on the FDDI Network Wide input screen.

[7] PERCENTSYN can be found on the FDDI Network Wide input screen.

Each station has a FIFO Link Layer queue, whose capacity (Station Specific input parameter LINKQUESIZE) is measured in terms of frames, which holds frames which are ready for transmission. Frames are transferred from the Network Layer to the Link Layer queue according to priority. If the frame at the head of the Link Layer queue is not eligible for transmission, it will block all other frames from transmission.

When a station receives a usable token (i.e. if the frame at the head of its transmission queue is eligible for transmission), then it captures the token, transmits a user-specified number of IDLE OCTETS, and then transmits frames as long as its timers permit. A new token is appended at the end of the station's data stream. If the token is not usable when it arrives at a station, the station simply repeats the token. Each frame that is transmitted propagates completely around the ring and is removed by the station which sent it. The destination station copies the frame, if there is sufficient room in its Network Layer receive buffers, and passes it directly to the Network Layer. If there is insufficient room to receive the message, the destination station returns a NACK to the sending station by setting the Frame Copied symbol on the frame. The sending station will then retransmit the frame at a later time.

When a station captures a usable token, the following specific timer-related actions are taken:

1. If the token is on time, i.e. if the station's Late Count flag is 0, then the TRT value is placed in THT and TRT is reinitialized to 0. Both synchronous frames and asynchronous frames may be transmitted.

2. If the token is late, i.e. if the station's Late Count flag equals 1, then the Late Count flag is cleared, but TRT is not reset. In this case only synchronous frames may be transmitted.

3. A station's THT is activated during the transmission of Idle Octets which precedes transmission of its data frames.

4. Transmission of a synchronous frame may be initiated only if sufficient synchronous bandwidth remains to complete its transmission.

5. A station's THT is activated during transmission of asynchronous frames. Transmission of an asynchronous frame may be initiated only if its priority is high enough, relative to the station's current THT value. The Network Wide user parameter PRIORTHRESH (priority threshold increment) is used to make this determination. If (framepriority)*(Priorthresh) >= THT, then transmission of the frame may be initiated. This check is made prior to transmission of each asynchronous frame.

6. When either the station's THT or its TRT expires, transmission of the current frame is completed and the token is passed to the next station.

3.5 Physical Layer

3.5.1 Star*Bus Physical Layer

The stations are connected by a fiber optic cable pair to a central, passive fiber optic star coupler. The Station Specific input parameter ARMLENGTH defines the length of the cable pair.

3.5.2 FDDI Physical Layer

The stations are connected in series to form a ring. The distance from each station to its downstream neighbor (toward which it passes all data in the flow around the ring) is the Station Specific input parameter NEXTDISTANCE. Internal station delay is given by the Network Wide input parameter ELASTICDELAY (elasticity buffer delay).

Chapter 4

Running LANES 3

The network model provides the user with a method to input simulation parameters and to archive results. LANES 3 has a series of screens that run the model.

The screens are arranged in a tree structure (see figure on page 17). They start with the Main Menu and progress to the three major branches; Input Parameters, Simulation, and Analyze. Depending on the protocol selected, each data entry screen performs a distinct function, or a related set of functions. A more complete description of each screen can be found in section 4.4 Screen Descriptions.

To make use of the screens first enter data in the fields that require user input, next press the ESCAPE key to move the cursor to the command line at the bottom of the screen. The command line is where the user selects commands for further processing of the data. Select a command by typing either the command or the first letter of the command and pressing the RETURN key. The network model then automatically retrieves or processes information on the screen and makes corresponding changes to the database.

Simple editing of user input is performed using the network model's screens and the simulation program. Some typographical errors and errors of type clash (a character where a number is required) are detected at this point.

Most of the parameters the user enters and most of the summary results produced by the simulation are stored in the LANES 3 relational database. User-provided descriptions of executed simulations are stored along with the simulation's results, so when a significant experimental result is produced, earlier results can be retrieved to provide a comparison. Another advantage of storing input values in a database is that the input values can be reviewed and modified to form the basis for new input parameters without re-entering the data.

4.1 The LANES 3 Computer Environment

LANES 3 uses two proprietary software packages and thus licences must be satisfied to enable porting to other systems. The model runs on a VAX minicomputer using the VMS operating system. The LANES3 account on PLUTO is the public account devoted to running the LANES 3 model and may be accessed remotely.

Some of the features of the LANES 3 network model, particularly those which deal with graphics, may not be effective on some types of remote terminals. Terminals or terminal emulations compatible with the VT100 series work well.

To start the network model in the demonstration account, log into the PLUTO VAX account, "LANES3". One user at a time may use the demo simulation set up for this account. The login screen will display PLUTO user messages, and will be followed by a line with a $ prompt. Type LANES3 after the prompt, so the screen looks like:

```
$lanes3
```

Upper and lower case characters may be used interchangably after the $ prompt, but all data entry in the network model is case sensitive and must be lower case.

By typing "lanes3" after the prompt, the Main Menu of the network model appears. This screen is the entry point to all of the other features of the network model. Each feature can be reached by moving from screen to screen. Refer to the figure on page 17 of the network model's screen structure to better understand the menu sequence. When finished with the screens, the user will return in reverse order of the calling sequence.

To use the model the user retrieves the screens to enter or change data, simulate and review results. The user chooses from among the options, on the bottom line of the screen, to execute the commands that affect the database or retrieve data to the screen. The commands may be performed in any order, and may be repeated in any order.

4.2 Using INGRES Screens

The screens the user sees and the database where information is stored are the part of LANES 3 that uses INGRES, a relational database from Relational Technology, Inc.

One important note is that a CONTROL Y or CONTROL C must never be used! Using CONTROL Y or CONTROL C causes INGRES to fail in a way that corrupts data when more than one user is running the network model and can cause the database to fail if one user is using INGRES.

The format of a screen includes a COMMAND LINE (list of commands available to the user located at the bottom of the screen) and the area of the screen located above the command line used for data entry. The data entry part of the screen displays fields that the user may access to review or enter data.

To select a command the cursor must be on the command line. Pressing the ESCAPE key moves the cursor to the command line. To select one of the commands the user types the command (or the first letter of the command) followed by pressing the RETURN key. To move the cursor back to the input area of the screen press the

RETURN key.[1]

Use the TAB key to move between data fields on the screen. Data can be entered in any data field the TAB key can access. To tab backwards use CONTROL P. Pressing the RETURN key when the cursor is on a data field will also act as a TAB. If the RETURN key is pressed without entering data the data field will default to a space if it is a character field or zero if it is a numeric field, and the cursor will move to the next data field on the screen.

The user must enter numbers in numeric fields, and characters and numbers in character fields. If an incorrect entry (for example: a character in a numeric field) is typed, an error message will appear and the user will have the opportunity to reenter the correct field type.

If the user detects a typing error before a command is entered the error may be corrected by moving the cursor to the field that contains the error and typing over it.[2]

In some cases a message appears at the bottom of the screen which either informs the user that some action has been taken, or informs the user of an error. Errors of a serious nature are likely to be somewhat incomprehensible. Errors in data entry can usually be corrected by deleting the data and reentering it. If there are extensive errors in data entry it may be easier to return to the Main Menu and start over. In practice we have encountered relatively few unrecoverable errors from use of the INGRES screens.

4.3 LANES 3 Database

All user input is stored in the database in table format. The user decides which simulations are stored in the database. The data in the database may be added, modified, deleted, or retrieved. Data entry and modification is done before the simulation is run. Retrieval (and optional deletion) of results in Analysis follows the data entry process.

A quick way to enter parameters is to retrieve the information from an existing simulation and copy it. A special function called "dup_input" is provided and can be found on the "input" screen. To use the "dup_ input" function, the user first types in the name of the new simulation if the simrunname wasn't passed from the Main menu. Next the user types in the name of the existing simulation to be copied. The parameters may be edited after they have been copied from the existing simulation to the new one.

[1] Usage tip: if no data entry is going to be done on a screen the user may want to press the ESCAPE key whenever a new screen appears. This will move the cursor to the command line and menu commands can be immediately selected by typing the first letter of the command. An extra ESCAPE where one is not needed does not affect the system.

[2] If the user decides that the whole screen has been incorrectly modified they may refresh the screen before an editing command (add, modify, delete) has been entered on the command line by selecting the retrieve command.

For simulations with new characteristics the user may want to enter data field by field. All input data is stored in the database identified by the simulation name (simrunname) and by the simulation protocol (protocol). Both are needed to identify a particular simulation run.[3]

To begin setting up the simulation first enter the input parameters on the screens then use the "add" command to append the data to the database or the "modify" command if existing values are being changed. After data has been appended or updated, the simulation retrieves the data it needs for inputs from the values stored in the database tables. The database table NETINPUT contains parameters which apply to the entire simulated NETwork. The database table STAINPUT contains parameters applying to the specifications of individual STAtions. Another table called SIMINFO contains user comments and the control information, including simulation run times.

When the simulation is finished it produces output files ZZNET.DAT, ZZNODE.DAT and *SIMRUNNAME*.TRACE. The first two files contain summary results for the entire network and for the individual stations and are used to produce reports. The files ZZNET.DAT and ZZNODE.DAT are temporarily stored in the main directory. The third file, *SIMRUNNAME*.TRACE, contains an event trace for the period defined by the user. See section 4.6 Interpreting Trace Output, for more information regarding the file *SIMRUNNAME*.TRACE. All of the files are stored in the subdirectory [lanes3.oldfiles] and are automatically named using the simulation's simrunname (example: for a simulation named "DEMO", ZZNET.DAT becomes DEMO.ZZNET when it is stored in the [lanes3.oldfiles] subdirectory and ZZNODE.DAT becomes DEMO.ZZNODE). The files used with Star*Bus become: DEMO.FZZNET, DEMO.FZZNODE, and DEMO.FTRACE.

These files are not automatically loaded into the database for two reasons. First, a particular simulation may be run to examine details of the Trace only. The Trace cannot be loaded into the database. If the user is not interested in the summary results then time can be saved by not loading ZZNET.DAT and ZZNODE.DAT into the database, and looking only at the file *SIMRUNNAME*.TRACE. Second, if a run has not succeeded in producing a specific experimental result there may be no purpose in storing the results. If the user wishes to use the Analysis screens or any INGRES function to process the output, the results need to be loaded via a simple command on the Analyze Screen. The Analysis screens only retrieve data, they do not modify it. The user may run Analyze functions without affecting stored data.

4.3.1 Field Naming Conventions

The database system requires that data entry fields be named in 12 characters or less. Since many fields have complex names they are abbreviated in some places. For example, "Simulation Run Name" becomes "simrunname" and "Elasticity Buffer

[3] A Star*Bus version and a FDDI version of the model can be run with the same simrunname.

Delay" becomes "elasticdelay". The output fields tend to be in a simple code where "av" = "average", "max" = "maximum", etc. See Glossary for more complete definitions.

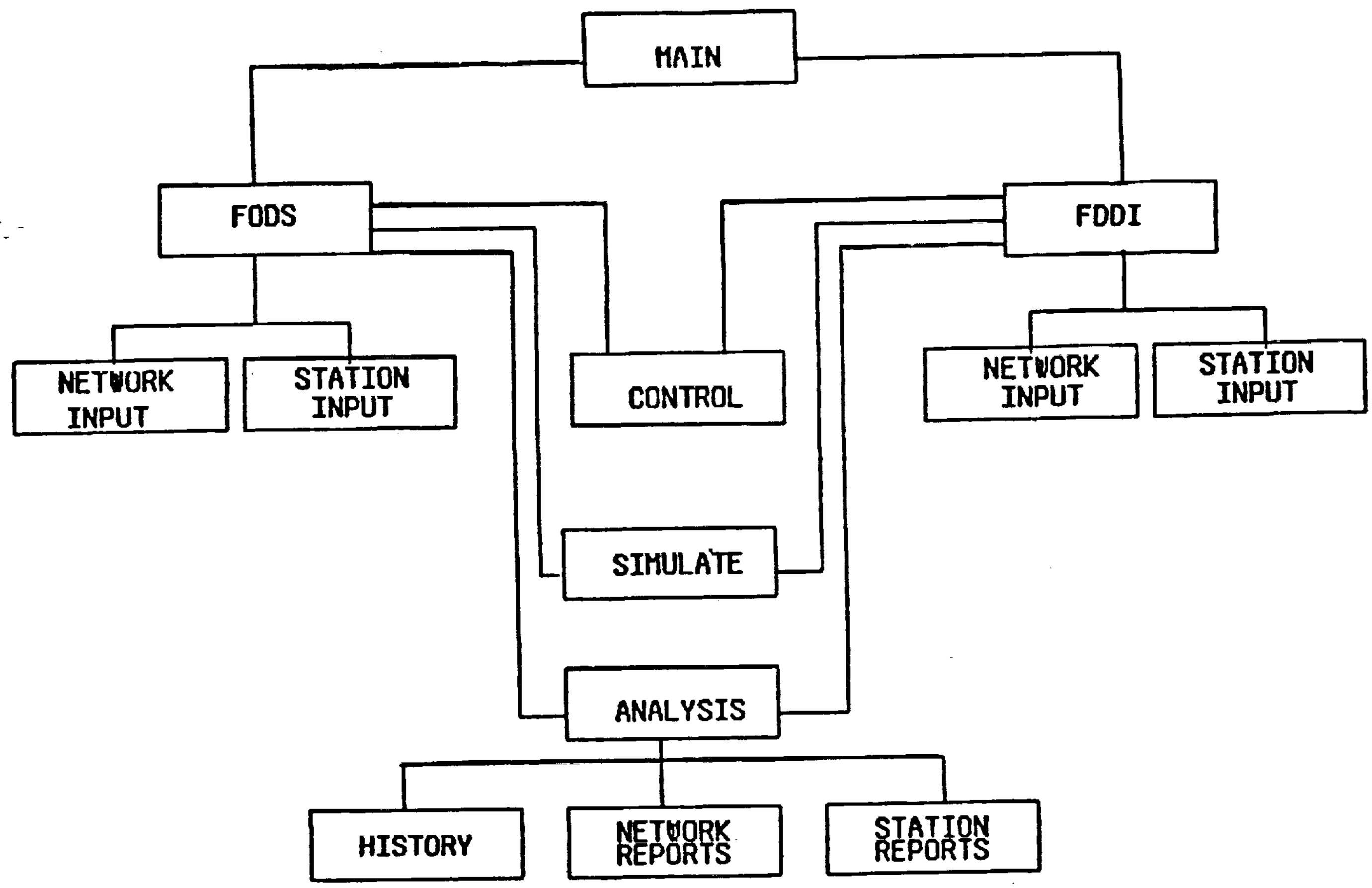

Figure 4.1: LANES 3 Network Model Screen Hierarchy

4.4 Screen Descriptions

4.4.1 Main Menu

This menu screen presents the user with ways to access the part of the system of interest. A menu screen is a method of putting other screens in a familiar context.

The data entry fields "simrunname" and "protocol" are the identifiers which apply to one unique set of inputs and the results generated from that simulation. Since a particular set of inputs always produces the same output, "simrunname" and "protocol" are the keys used by the database to describe an entire simulation. Each simulation whose output is to be stored in the database, must have a unique "simrunname" if the same protocol is to be used.[4] A new simrunname does not have to be entered for simulations which are run only to generate a Trace.

The data entry field "protocol" on the Main Menu allows the user to select which type of protocol is to be simulated. To simulate the star local area network protocol type "fods". To simulate the ring local area network type "fddi". The network model will then be directed to the selected type of protocol.

The commands on screen Main Menu are:

- "help" Access screens which have information about running the network model.
- "input" Enter or review the input parameters for simulations.
- "simulate" Access the screen where the user runs the simulation. This command is used after input data has been added or modified.
- "analyze" Access screens where output run results can be loaded and reviewed.
- "exit_application" End the execution of LANES 3. The user is returned to the VAX operating system level. After exiting user may review the *SIMRUNNAME*.TRACE file, located in the subdirectory [lanes3.oldfiles], run LANES 3 again or logout by typing "bye" after the $ prompt. Any number of simulations can be run without leaving the network model screens.

4.4.2 Input Parameters

This Input Menu screen gives the user direct access to the data entry screens for Network Wide, Station Specific and control inputs for the model. With the screens that can be reached from here, the user has the option to do one of three things:

1. enter all new simulation input data.

[4] A simulation using the FDDI protocol can have a simrunname "demo" and a simulation using the Star*Bus protocol can have a simrunname "demo", but two simulations using the same protocol cannot have the same simrunname.

2. review previous simulation input data.

3. make a copy of previous simulation input data under a different name.

The command options are:

- "dup_input" Make a copy of previous simulation input data under a different name.

  To duplicate an existing simulation's inputs type in both the new simulation's name (simrunname) and the old simulation (old simrunname). Press the ESCAPE key and type "dup_input" or "d" on the command line then press the RETURN key. All the database input records for the old simulation will be automatically duplicated into the database records with the new simrunname. Since all the information of a previous simulation has been duplicated, the user may continue to select the "network" command and change only the fields which require new values.

- "network" Go to screen for Network Wide parameters.

- "station" Go to screen for Station Specific parameters.

- "control" Go to screen for simulation control parameters, and for general comments.

- "end" Return to Main Menu.

4.4.3 Enter or Review Network Wide Parameters

This screen can be used for four things:

- "retrieve" To review previously entered data for the current simrunname, or to review data of any existing simrunname.[5]

  To see data which was previously entered, type in "simrunname" press the ESCAPE key and type "retrieve" after the command line prompt. Then press the RETURN key to execute the command.

- "add" To enter all new data. (When the default input data does not already exist. Data would already exist if a "dup_input" was used)

- "modify" To update the simulation inputs shown on the screen with the values that have been modified by the user.

[5] If the user needs to locate a simrunname they should refer to the "History" screen. The History screen displays information regarding the simulations currently stored in the database.

- "query" Makes use of INGRES' Query-By-Form command. See APPENDIX-A for more details.
- "end" Returns to input parameter menu screen.

The value of any field can be changed by TABing to that field and typing the new value over the one currently displayed. When all fields have been changed, press the ESCAPE key and type "modify" or the letter "m" on the command line then press the RETURN key to execute the command. The database now contains the revised values.

To enter an entire set of new values enter data in every field which the user wishes to be non-zero, and enter command "add". The user will be notified when the values are accepted. If a simrunname of this name already exists the user will be prompted to type in a different value for simrunname.

4.4.4 Enter or Review Station Specific Parameters

This screen allows the user to change the input parameters for each station. This screen uses two command lines, where a second, entirely different, command line is presented to the user in response to an initial "retrieve" command. Since "retrieve" causes the retrieval of database records for each station, and only one station is displayed at a time, commands are needed to look in turn at each station. This second command line is called a SUB-MENU.

When finished using the commands on the sub-menu, return to the Input menu screen commands by typing "end".

- "retrieve" Retrieves all the stations for simrunname and displays the first station on screen along with a sub-menu with the following commands:

  o "show_next_station" Show values for the next station.
  o "modify" Change values for the stations specified. (see "add")
  o "end" Return to the station specific parameter menu screen.

- "one_retrieve" Prompts for the node number of the station that the user would like to review. Displays the data for only that station.
- "add" Type in the new values leaving the nodenumber field blank. After all the new values are correctly entered, press the ESCAPE key, type "add" at the command line then press the RETURN key to execute the command. A message will be displayed prompting the user to enter the node numbers (station numbers) [6] that the new values will affect. If duplicate node numbers exist then the old values will be overwritten by the new values.

[6]Use a space or comma between node numbers (station numbers) to act as delimiters, e.g. 1,3,17 or 1-5,13

- "query" Makes use of INGRES' Query-By-Form command. See APPENDIX-A for details.
- "end" Return to input parameter menu screen.

4.4.5 Control (Enter or Review Parameters)

Two types of data are presented on this screen, values which control the simulation, and free form fields where the user can enter comments. It is suggested for archiving purposes and the convenience of other users that brief entries be made in the comment fields (maximum 57 characters per line).

- "retrieve" Retrieve existing values in the database for the simrunname shown on the screen.
- "add" Add new control values to the database. This is used when no values exist in the database already, and all new information is being added by the user.
- "modify" Modify existing values for the simulation shown on the screen with the values currently shown on the screen.
- "query"Makes use of INGRES' Query-By-Form command. See APPENDIX-A for details.
- "end" Return to the input parameter menu screen.

4.4.6 Simulation Start Screen

From this screen the user begins the simulation. Time required to run the simulation depends on the runtime of simulation, the number of messages generated, the number of stations modeled and the amount of congestion on the NASA/Ames VAX computer.

Several messages appear on the screen during the interactive simulation process. There is a series of messages generated by the database handling routines. These messages will indicate the number of rows copied from the database to a file read by the program that runs the simulation. The message, "(X rows)", X being the number of stations in this simulation, and another message, "(1 row)", will appear on the screen. If the values displayed on the screen do not match the input parameters, the user may consider that the incorrect number of stations has been entered, or that the input is otherwise incomplete. If the simulation is being run in batch mode, the user exits the network model. If the simulation is being run interactively it must be allowed to complete normally.

When running the simulation in interactive mode, the simulation start screen briefly reappears after the simulation input files have been created. Then the simulation starts and the message "Initialization for simulation has started." appears. After a period of about a minute the following message appears, "Initialization complete, simulation Z commencing now.", where Z is the name of the simulation being run. Shortly after the simulation passes each 5000 ms., a message "Simulation has passed Y ms." appears, where Y is the simulation time which has recently been reached. At the end of the simulation data appears which is primarily for debugging purposes. The user may be interested in the final part of the process which displays the results for the simulation. Using "No Scroll" or "CONTROL S" temporarily halts the scrolling of data on the screen. The user will then be able to review the information display on the screen. Pressing "CONTROL Q" will resume scrolling. The final message for a successful simulation run is, "Simulation has completed. Program will end momentarily."

When the simulation has finished running, the Simulation Start Screen will reappear. The user may return to the Main menu and select the "Analyze" command to review the results generated by the simulation.

All other messages are either warning or error messages. Errors halt the simulation immediately. Warnings about field overflows, which do not stop the simulation, mean that some information is being lost in the *SIMRUNNAME*.TRACE file or in the output statistics. Field overflows in the Trace do not affect the integrity of the output statistics. It is important, if the user does not immediately understand the error, to keep the text of the message.

LANES 3 also provides a way to run a simulation in batch mode, rather than interactively. Running the simulation in batch mode allows the user to log off the system while the simulation is runnning. Batch jobs run at a lower priority than interactive jobs.

Any number of simulations can be submitted for batch processing. For each simulation that is to run in batch mode, enter the simrunname on the Simulation Start screen and select the "create_batch_files" command by pressing the ESCAPE key, typing "c" then pressing the RETURN key. The Simulation Start screen will disappear briefly then reappear after the batch file creation process has finished. The user may repeat this process for any number of simulations.

To run the simulations created in batch mode, exit the LANES 3 network model[7] and type "submit LANESBATCH.COM" after the VMS prompt ($). The output files and trace are created and renamed the same way they would be if the simulation had been run interactively. The file LANESBATCH.LOG will be created. It contains the information that would have been output to the screen if the simulation had been run interactively. The file LANESBATCH.LOG also contains VAX performance figures and any error messages that occured during the simulation. It is recommended that batch processing be scheduled to run after 5:00 p.m.(PST).[8]

SLAM files cannot be opened in a shared mode therefore, LANES 3 cannot run

[7] Return to the Main menu by selecting the "end" command. Then select the "exit_application" command on the Main menu to return to VMS.

[8] The VMS command to run the simulations created in batch mode after 5:00 p.m. is: "submit/after=17:00 LANESBATCH.COM"

simulations interactively and in batch mode at one time nor can LANES 3 run more than one simulation interactively at the same time.

- "run_simulation" Use existing input data stored in the database to run a simulation.
- "create_batch_files" Allows the user the option of running the simulation in batch mode rather than interactively.
- "end" Exit this screen, return to the main menu.

The screen will not return any error messages if the simulation completes successfully. Possible causes of a simulation failure are:

1. Missing files. FOR008.DAT (for both Star*Bus and FDDI), RING1.DAT, RING2.DAT, RING30.DAT, RING31.DAT, RING32.DAT, must exist for FDDI simulations; FODS1.DAT, FODS2.DAT, FODS3.DAT, FODS30.DAT, FODS31.DAT, FODS32.DAT must exist for Star*Bus (FODS) simulations.
2. User does not have enough quota in account to store all the output. This can occur if the user requests a Trace output.
3. Bad input data. Check to see that all fields are correctly specified, that there is exactly one set of parameters for every station.
4. Unusual data input. The simulation is not extensively protected against negative numbers, very large numbers, or any numbers generally which reflect situations which would not occur in a real hardware configuration.[9]

Once the simulation is complete the user may proceed at once to analyze the results.

If the user has left the network model to view Trace they may return to the network model at this point by typing "lanes3" at the VMS prompt. No simulation with Sub-Periods may be loaded into the the database, but the same simulation can be re-run without Sub-Periods and placed into the database.[10]

4.4.7 Analyze

This screen is the departure point to all analysis of the output data. The data from the simulation is loaded into the database at the user's discretion with the command "load_sim". Reports and screens will use this information.

The "history" screen is a method for reviewing and controlling which simulations are currently stored in the database. After "load_sim" has been executed, the user may

[9] Please note: user should not enter data more than 2 orders of magnitude larger than the examples given in the demo.

[10] The Sub-Periods data entry field can be found on the Control input screen.

select either the "network_reports" or "station_reports" command to review results via query screens or printed reports.

- "load_sim" Take the output from the latest simulation and move it from the files created by the simulation to the database.
- "history" Go to the History screen.
- "network_reports" Go to the Network_Reports screen.
- "station_reports" Go to the Station_Reports screen.
- "end" Return to the Main menu.

4.4.8 History

This screen performs a retrieval on every simulation in the database and shows an easily identifiable extract of their information. The data fields are limited in size, so only the first part of some fields appear. Other simulations are shown by scrolling the screen with CONTROL J and CONTROL K.

This screen also provides a method to delete a simulation, or to delete only the output for a simulation.

- "run_delete" Delete all records for this simulation from the database. To use this feature place the cursor at the start of the row of the simulation to be deleted, press ESCAPE key, and enter command.
- "output_delete" Delete only the output records for this simulation. Place the cursor on the command line by pressing the ESCAPE key, type "o" (oh, not zero), then press the RETURN key to execute the command. Once the output for a simulation has been deleted another simulation of the same simrunname can be run, with the same input if desired, and its output loaded into the database. If only the output for a simulation is deleted the input information is still kept for use in the database.
- "end" Return to the Analyze screen.

4.4.9 Reports

Network Results

This screen allows the user the option of sending the network-wide results to a printer or browsing the information directly on the screen.

- "print_network_results" Sends a copy of the Network Wide results report to the QMS laser printer located at NASA/Ames Research Center.

- "query_network_results" Makes use of INGRES' Query-By-Form command. See APPENDIX-A for details.
- "end" Return to Analyze menu screen.

Station Results

This screen allows the user the option of sending the station-specific results to a printer or browsing the information directly on the screen.

- "print_station_results" Send a copy of the Station Specific results report to the QMS laser printer located at NASA/Ames Research Center.
- "query_station_results" Makes use of INGRES' Query-By-Form command. See APPENDIX-A for details.
- "end" Return to Analyze menu screen.

4.5 Analyzing Results

Analysis follows the running of the simulation. LANES 3 has provided screens and on-line reports to review results, as well as the Trace file. These reports provide the basic tools to review results, and to make simple comparisons.

More sophisticated analysis can be made by use of INGRES database tools, or by processing the output files ZZNET.DAT and ZZNODE.DAT. To do more complicated analysis outside of the scope of the LANES 3 network model interested users should contact NASA/Ames Research Center.

4.6 Interpreting Trace Output

An optional feature is the production of a 2000 microsecond Trace of events which occur during the simulation. The trace is stored in a file called "*SIMRUNNAME*.TRACE". It appears in the subdirectory [lanes3.oldfiles].

It contains a sequential listing of significant events in the simulation. Included is the time of occurrence, the station (node), and the event. Events include receipt of a message, or acknowledgment of a message, or the beginning or ending of a Backoff period. The expiration of the TRT, messages, types and sizes are also noted. The value of the TRT and the amount of space in a receiving station's Network Layer buffer are also included where relevant.

The starting time, which will be reported by Trace, is set on the Control screen with the parameter STARTTRACE. The events of a simulation are recorded in the file *SIMRUNNAME*.TRACE for a period of up to 2000 microseconds. The file may be of considerable length (tens of pages). It may be reviewed outside the simulation. To review the file after the simulation has run, exit the network model and view the file either by using the VMS editor, or by typing:

$ TYPE *SIMRUNNAME*.TRACE

in response to the VMS prompt.

This file is not used by the network model so the user may change, delete, or use this file in any way without affecting the analysis module. The file may be be several pages long so any unneeded files should be deleted.

If the user does not wish to use the Trace, the minimum amount of information will be written by specifying a starttrace time equal to the runtime on the Control screen.

Time units are measured in microseconds. Messages are numbered chronologically as they are generated. Frame numbers are generated as the message is broken down into frames in the Network Layer.

Any value which has more digits than the space allowed for it in a trace statement will have "******" in the field in place of a number. The number of stars is the greatest number of digits, excluding decimal point, which can be printed at that point.

Chapter 5

Running a Demo

Every user will want to investigate the preceding chapters of the users' manual in some detail. However to quickly understand the system the user may prefer to follow the steps of running the demo. Running the demo involves very little data entry on the user's part, and will give the user an idea of how quickly a simulation can be set up.

First, log into the LANES3 account on the Pluto VAX at NASA/Ames Research Center.[1] After the VMS $ prompt at the beginning of the line type:

lanes3

The Main menu of the network model immediately appears on the terminal screen. The cursor is located at the data entry field "simrunname". For purposes of running the demo the user can type any 10 character name for a simulation after the prompt (test, demo23, etc.). Next type in the protocol that is to be simulated in the field just below the simrunname field. Type either "fddi" or "fods". The default protocol is fddi.

At the bottom of the screen is the COMMAND LINE. This line presents the user with options available for processing the information on the screen, or for accessing information in the database. To perform a command press the ESCAPE key, then type the first letter of a command (or the entire command), then press the RETURN key.

Begin by pressing the ESCAPE key and typing "h" for help after the command line prompt at the bottom of the screen. Press the RETURN key to execute the command. This command retrieves a series of on-line help screens. These screens will give the user more complete information regarding hands-on use of the simulation.

Now type "e" (exit) to return to the Main menu. The user is back at the beginning of the network model. The LANES 3 network model is arranged in a "tree" structure. It is possible to traverse the tree in a specific order either in a forward or backward direction.

---

[1]See section 4 Running LANES 3 for information regarding access to the LANES3 account.

For the demonstration we have entered input parameters which the user may copy. The name that was typed in the field "simrunname" on the Main Menu will be the identifier for the simulation the user is about to run. Each simulation using the same protocol must have a unique identifier (simrunname).

To move to the command line press the ESCAPE key. The user may move back and forth between the command line and the data entry screen by pressing the ESCAPE key, or by pressing the RETURN key. See the help screen "cursor_help", which can be accessed by typing "help" in the Main menu, for more details.

Type "i" for inputs. The Input screen acts as a menu screen for selecting which type of input data is to be entered. Notice that the name (test, demo23, etc.) which was typed in the simrunname field on the Main menu screen appears here also.

To duplicate information which we have entered in simulation "demo" type demo in the field called "old simrunname". Pressing the ESCAPE key moves the cursor to the command line. Type "d", the first letter of the command "dup_input", then press the RETURN key. After a few seconds a message will appear on the command line at the bottom of the screen indicating that the information from the old simulation has been successfully duplicated. If the message indicates that a simulation of that name already exists, type in another simrunname. After inputs have been successfully duplicated they can be reviewed, and modified.

Next the user should select the "network" option from the command line on the Input menu screen. The Network Wide input screen will appear. Use the TAB key to move the cursor between the data entry fields on the screen. To modify the values on the screen, just type over the values that appear. After the Network Wide parameters have been modified on the data entry portion of the screen, the user should select the "modify" command to replace the existing Network Wide parameters with the new ones in the database. The user can now select the "end" command from the network input screen to return to the Input menu screen.

After the Network Wide parameters have been entered, the user should enter the Station Specific parameters for each station on the network. The Station Specific input screen is called by selecting "station" from the Input menu screen's command line.

To make modifications to the Station Specific parameters, first select "retrieve" from the commands listed. This command retrieves the parameters for the first station in the network. Modifications to the existing parameters can be made directly on the data entry portion of the screen. When modifications have been to made to the station, select the "modify" option to replace the old Station Specific parameters with the new ones in the database. The "show_next_station" and "modify" commands will enable the user to modify the rest of the stations in the network. When all stations have been modified the user may return to the Input menu screen by selecting "end".

By using the Control Screen the user may change the values in the control input parameter fields. The user may enter some data in the comment fields and select the "modify" command to update the database. The information is stored permanently in the database for later review and use. Use the "e" (end) command to return to the

Input menu screen.

Exit the Input menu screen and return to the Main menu screen by pressing the ESCAPE key and typing "e" or "end", then pressing the RETURN key. At the Main menu type "s" for simulate. The Simulation screen performs the simulation whose inputs the user has just entered, or the user may enter the simrunname "demo" and run the demo simulation by selecting the "r" or "run" command.[2] The output from the demo simulation has already been stored in the database and the results may be reviewed by the user.

Return to the Main menu screen by typing "e" at the command line. Back at the Main menu type "a" for analyze. If the user is running their own version of the simulation instead of "demo", after the simulation has been run successfully, the "l" (load_sim) command should be used to retrieve the results of the simulation into the database.

After the simulation has been loaded into the database, select either "n" (network_reports) or "s" (station_reports) command to retrieve the report menu screen for Network Wide or Station Specific results. When the report screen appears, select the "query_network_results" (or "query_station_results) to browse through the information generated by the simulation. After reviewing the results the user may now return to the Main menu and exit the network model using the "e" (exit) command.

The user has now gone through almost the entire process of running a simulation. What remains is to learn how to modify the input data, to run the Trace facility, and use the more sophisticated analysis techniques. Refer to relevant sections of this user manual for information. The user should read in detail the instructions for running the network model, and other sections as needed.

[2]See section Simulation Start Screen, page 21, for information on the messages that will appear when the simulation is running.

Chapter 6

Appendixes

6.1 -APPENDIX A-

6.1.1 Using INGRES QBF (Query By Forms)

The Query By Forms [1] option offered by INGRES provides a way for users to access information in the LANES 3 database directly through the screens provided by the LANES 3 network model. On-line help messages and guidance are constantly available to explain the use of the commands listed at the bottom of each query screen.

The query screens in the LANES 3 network model are set up to retrieve data from the database tables "netresult" and "staresult". To access the query screens first select the "query_network_results" (or "query_station_results" if using the Station Specific Results screen) command before specifying a query. The screen will disappear and INGRES generated messages will appear at the bottom of the screen. The query screen will reappear, (see figure on page 49) and the user can begin to enter specifications for the data they wish to review.

The retrieve function consists of two states, the QUERY state and the GO state. While in the QUERY state the user specifies the query by filling in the fields on the screen with the values that are to be retrieved. After the user fills in the screen, QBF enters the GO state and retrieves the specified rows from the database. To enter the GO state, after specifying a query, the user presses the ESCAPE key, types "g" or "go" at the command line, then presses the RETURN key.

If INGRES is unable to locate the rows qualifying the user's query, QBF displays the following message at the bottom of the screen.

NO ROWS FOUND

QBF then returns to the QUERY state for the next query.

[1] Reference: INGRES QBF (Query By Forms) USER'S GUIDE Version 2.1.

If there is more than one row of data specified by the query, the user can press CONTROL F ("F" for forward) to retrieve the next row of data. When QBF can not find any more rows that satisfy the original query the following message will be displayed at the bottom of the screen.

NO MORE ROWS LEFT IN QUERY

Entering A Query

If all fields are left blank on the query screen, QBF assumes the user wants to retrieve all the rows in the database table (either netresult or staresult). The best way to retrieve only the values the user is interested in reviewing, is to fill in only the fields on the query screen that will identify the specific data.

Example: Query Screen Network Wide Results

simrunname: demo
protocol: fddi

Retrieves only the Network Wide results for the simulation named "demo" with protocol "fddi".

Comparison operators may be used when specifying a query. This is done by placing a valid INGRES operator in front of the value in the data field. The following are valid INGRES operators:

=	EQUAL
!=	NOT EQUAL
<	LESS THAN
<=	LESS THAN OR EQUAL
>	GREATER THAN
>=	GREATER THAN OR EQUAL

Example: Query Screen Station Specific Results

simrunname:demo
protocol:fddi
nodenumber: <= 4.

Displays the Station Specific output results for the simulation named demo using the FDDI protocol for the stations numbered 1, 2, 3, 4.

Fields containing values and no comparison operators are assumed to have an EQUAL operator.

Selecting the "end" option, at the bottom of the screen when the QBF session is complete, will return the user to the LANES 3 report screen. The user may then select the other options available on the report screens or return to the Main menu.

6.2 -APPENDIX B-

6.2.1 Features of Star*Bus and FDDI

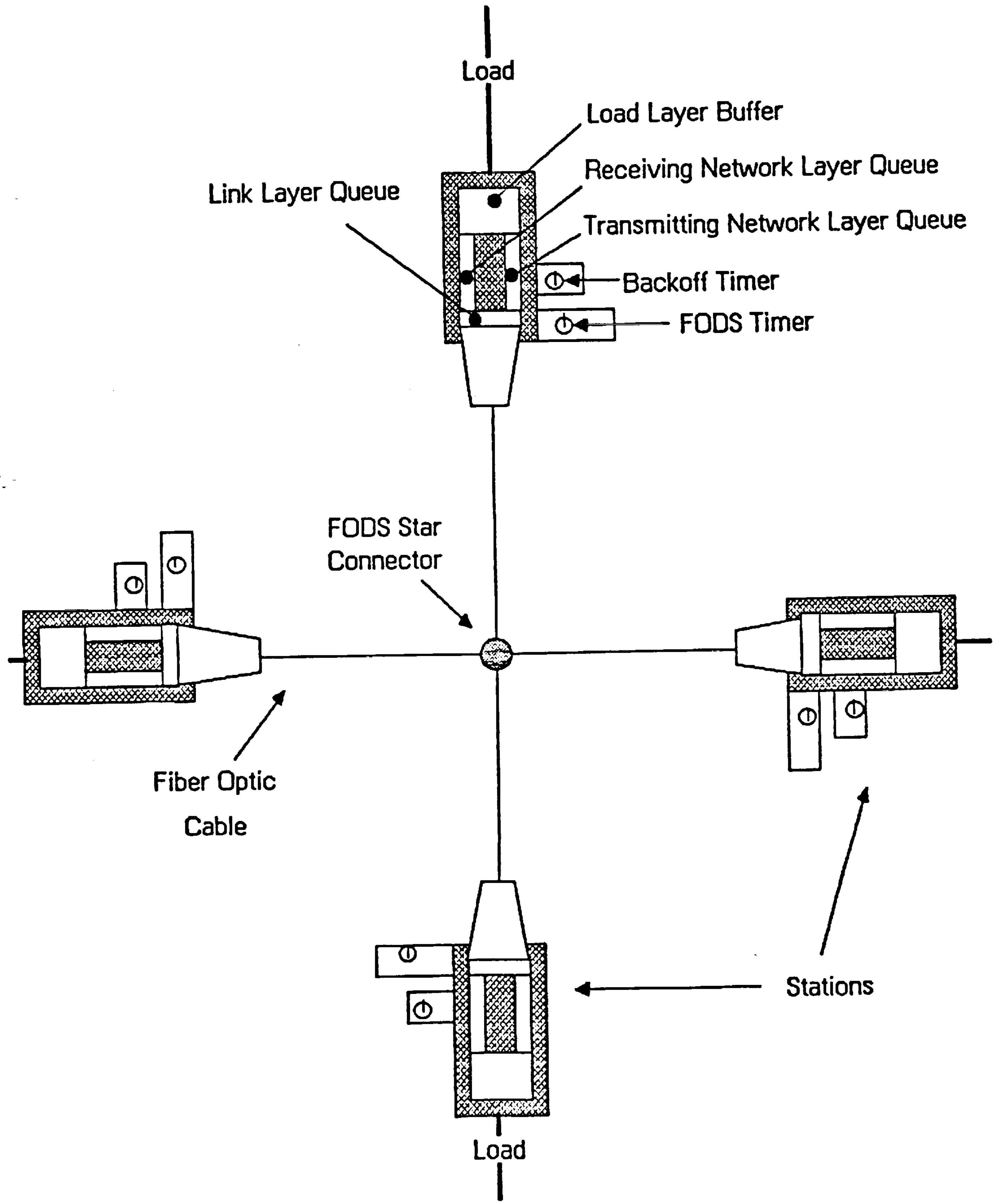

Figure 6.1: Features of Star*Bus Simulation

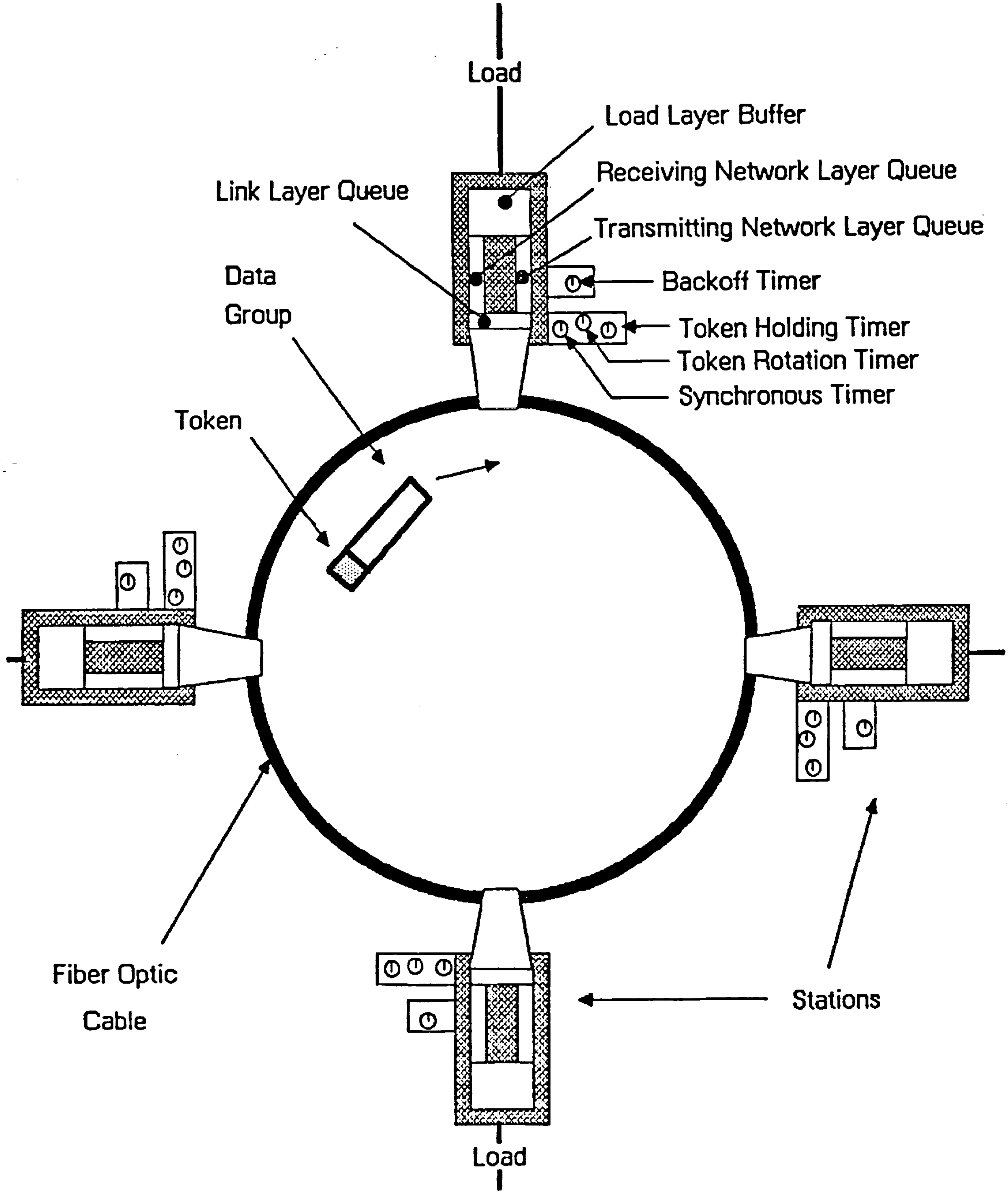

Figure 6.2: Features of FDDI Ring Simulation

6.2.2 LANES 3 Network Model Screens

```
---------------------------------------------------------------
|      Welcome to the LANES 3 Simulation                       |
|                                                              |
|   The Local Area Network Extensible Simulator (LANES)        |
|provides simulation and modelling capability of potential     |
|Space Station Data System configurations.                     |
|   This application can be used to model the physical and     |
|link layers of either a star topology local area network      |
|using the Fiber Optic Demonstration System (FODS) or a        |
|token passing ring local area network based on the Fiber      |
|Distributed Data Interface (FDDI) Token Ring Media Access     |
|Control. The same network layer and load layer are used for   |
|each.                                                         |
|                                                              |
|   To run a simulation using this application type a name     |
|   for the simulation after simrunname and follow the steps   |
|   below:                                                     |
|                                                              |
|   o Input       Enter data regarding network configuration,  |
|                 and simulation characteristics.              |
|                                                              |
|   o Simulation  Run simulation with existing input.          |
|                                                              |
|   o Analyze     Query results on screen or request           |
|                 printed reports.                             |
|                                                              |
|               simrunname:            (optional)              |
|               protocol:              (fods or fddi)          |
|                                                              |
|  help  input  simulate  analyze  exit_application            |
---------------------------------------------------------------
```

Figure 6.3: Main Menu

```
-----------------------------------------------------------------
|    Access Menu to Add and Edit Simulation Parameters          |
|                                                               |
|                              fods                             |
|                                                               |
|  From this screen user can reach all the system               |
|  parameters which can be modified to tailor simulation        |
|  characteristics.                                             |
|                                                               |
|         o  Parameters for Star Network                        |
|                                                               |
|         o  Parameters for Each Station                        |
|                                                               |
|         o  Control Parameters                                 |
|                                                               |
|                                                               |
|                                                               |
| If you wish to use an old simulation as the base for a        |
| new set of parameters, enter old and new name, and type       |
| command "dup_input".                                          |
|                                                               |
|                   simrunname:                                 |
|                   old simrunname:                             |
|                                                               |
| dup_input  network  station  control  end                     |
-----------------------------------------------------------------
```

Figure 6.4: Star*Bus Input Parameter Menu Screen

```
------------------------------------------------------------------
|                                                                |
|         Enter or Review Network Wide Parameters                |
|                                                                |
|    Enter parameters which apply to all stations or to          |
|    entire network below:                                       |
|                                                                |
|                        simrunname:                             |
|                        protocol: fods                          |
|                                                                |
|largestframe:     (octets)       numstations:                   |
|                                                                |
|headersize:       (octets)       xmissionrate    (megabit/sec)  |
|                                                                |
|signalspeed:      (meters/sec)   backofftime:    (usec)         |
|                                                                |
|bufferdelay:      (usec)         timeslotwidth:  (usec)         |
|                                                                |
|gaptime:          (usec)         rdmstepsize:    (usec)         |
|                                                                |
|maxtimestep:                                                    |
|                                                                |
|  retrieve   add   modify   query   end                         |
------------------------------------------------------------------
```

Figure 6.5: Star*Bus Network Wide Input Parameter Screen

```
-----------------------------------------------------------------------
|                                                                     |
|       Enter or Review Station Specific Parameters                   |
|                                                                     |
|                                   simrunname:                       |
|                                   protocol: fods                    |
|                                                                     |
|nodenumber                            class:                         |
|armlength:     (meters)               netrecevsize:   (octets)       |
|netxmitsize:   (octets)                                              |
|                                                                     |
|                                                                     |
|                   Interarrival Time    Message Size                 |
|                    Type Var 1 Var2    Type Var 1 Var 2              |
|synchronous                                                          |
|asynch priority 2                                                    |
|asynch priority 1                                                    |
|                                                                     |
|                                   Absorption Time                   |
|                                  Type Var 1 Var 2                   |
|         synchronous                                                 |
|         asynch priority 2                                           |
|         asynch priority 1                                           |
|                                                                     |
|  retrieve  one_retrieve  add  query  end                            |
-----------------------------------------------------------------------
```

Figure 6.6: Star*Bus Station Specific Input Parameter Screen

```
-------------------------------------------------------------
|     Access Menu to Add and Edit Simulation Parameters      |
|                                                            |
|                           fddi                             |
|                                                            |
|   From this screen user can reach all the system           |
|   parameters which can be modified to tailor simulation    |
|   characteristics.                                         |
|                                                            |
|          o  Parameters for Ring Network                    |
|                                                            |
|          o  Parameters for Each Station                    |
|                                                            |
|          o  Control Parameters                             |
|                                                            |
|                                                            |
|                                                            |
| If you wish to use an old simulation as the basis for a    |
| new set of parameters, enter old and new name, and type    |
| command "dup_input".                                       |
|                                                            |
|                  simrunname:                               |
|                  old simrunname:                           |
|                                                            |
|  dup_input  network  station  control  end                 |
-------------------------------------------------------------
```

Figure 6.7: FDDI Input Parameter Menu Screen

```
--------------------------------------------------------------------
|      Enter or Review Network Wide Parameters                       |
|                                                                    |
|  Enter parameters which apply to all stations or to                |
|  the entire network                                                |
|                                                                    |
|                       simrunname:                                  |
|                       protocol: fddi                               |
|                                                                    |
|largestframe:    (octets)     numstations:                          |
|                                                                    |
|headersize:      (octets)     xmissionrat:  (megabit/sec)           |
|                                                                    |
|signalspeed:     (meters/sec) numidleoctet:                         |
|                                                                    |
|elasticdelay:    (bits)       priorthresh:                          |
|                                                                    |
|toknrotation:    (usec)       percentsyn:                           |
|                                                                    |
|backofftime:     (usec)                                             |
|                                                                    |
|  retrieve  add  modify  query  end                                 |
--------------------------------------------------------------------
```

Figure 6.8: FDDI Network Wide Input Parameter Screen

```
----------------------------------------------------------------------
|        Enter or Review Station Specific Parameters                  |
|                                                                     |
|                        simrunname:                                  |
|                        protocol: fddi                               |
|                                                                     |
|nodenumber:                 class:                                  |
|nextdistance:   (meters)    netrecevsize: (octets)                  |
|netxmitsize:    (octets)    bandwidth:    (normalized to %)         |
|linkquesize:    (# of frames)                                       |
|                                                                     |
|                                                                     |
|                Interarrival Time    Message Size                    |
|                 Type Var 1 Var2   Type Var 1 Var 2                  |
|synchronous                                                         |
|asynch priority 2                                                   |
|asynch priority 1                                                   |
|                                                                     |
|                            Absorption Time                          |
|                           Type Var 1 Var 2                          |
|        synchronous                                                  |
|        asynch priority 2                                            |
|        asynch priority 1                                            |
|                                                                     |
|  retrieve  one_retrieve  add  query  end                            |
----------------------------------------------------------------------
```

Figure 6.9: FDDI Station Specific Input Parameter Screen

```
-------------------------------------------------------------
|       Enter or Review Control for Simulation              |
|      (simirunname and runtime must be entered)            |
|                                                           |
|                                                           |
|simrunname:             runtime:            (usec)         |
|                                                           |
|protocol:                                                  |
|                                                           |
|startstatcol:  (usec)  subperiods:          (usec)         |
|                                                           |
|starttrace:    (usec)  simrundate:                         |
|                                                           |
|primarypur:                                                |
|                                                           |
|comments1:                                                 |
|                                                           |
|comments2:                                                 |
|                                                           |
|comments3:                                                 |
|                                                           |
|sigresult:                                                 |
|                                                           |
|  retrieve  add  modify  query  end                        |
-------------------------------------------------------------
```

Figure 6.10: Control Parameter Input Screen

```
------------------------------------------------------------------
|                 Simulation Start Screen                        |
|                                                                |
|    The user defined input parameters are read from the         |
|    database with the simrunname specified.                     |
|                                                                |
|     When this screen returns permanently the simulation        |
|     is complete.                                               |
|                                                                |
|     Output appears in the default files, and may at            |
|     once be read into the database for analysis.               |
|                                                                |
|     Consult user manual on the relation between the            |
|     application files, database, and screens for               |
|     advanced use.                                              |
|                                                                |
|     -  To create files to run the program in batch             |
|        mode use command "create_batch_files"; do not           |
|        use command "run_simulation".                           |
|                                                                |
|                                                                |
|           Enter simrunname:                                    |
|                                                                |
|  run_simulation  create_batch_files  end                       |
------------------------------------------------------------------
```

Figure 6.11: Simulation Start Screen

```
--------------------------------------------------------------
|               Analyze Simulations Menu                     |
|                                                            |
|                                                            |
|     This screen is the top level analysis screen.          |
|     Analysis screens are used to review data, and to       |
|     add and delete information from the database.          |
|                                                            |
|                                                            |
|  Type "load_sim" to load the results from the last         |
|  simulation in the database.                               |
|                                                            |
|                                                            |
|                     simrunname:                            |
|                     protocol:                              |
|                                                            |
| load_sim  history  network_reports  station_reports  end   |
--------------------------------------------------------------
```

Figure 6.12: Analyze Simulation Menu Screen

```
-----------------------------------------------------------------
|        Review Simulations Currently in Database                |
|+----------+--------+----------+----------+---------------+     |
||simrunname|protocol|simrundate|comments1 |sigresult      |     |
||==========+========+==========+==========+===============|     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
||          |        |          |          |               |     |
|+----------+--------+----------+----------+---------------+     |
|                                                                |
|  run_delete  output_delete  end                                |
-----------------------------------------------------------------
```

Figure 6.13: History Screen

```
---------------------------------------------------------------
|            Query Network Wide Results                        |
|            simrunname:         protocol:                     |
|numrotation:   totofferedld:   throughput:   utilzation:      |
|msgdelv:       synfrmdelv:     asyn2frmdelv: asyn1frmdelv:    |
|frmdelv:       ttxr:           qf:                            |
|                      Average Maximum Minimum Std Deviation   |
|load to load delay                                            |
|             synch                                            |
|             asyn2                                            |
|             asyn1                                            |
|         all types                                            |
|link to link delay                                            |
|             synch                                            |
|             asyn2                                            |
|             asyn1                                            |
|         all types                                            |
|load to link delay                                            |
|             synch                                            |
|             asyn2                                            |
|             asyn1                                            |
|         all types                                            |
|link to load delay                                            |
|             synch                                            |
|             asyn2                                            |
|             asyn1                                            |
|         all types                                            |
|FDDI ONLY: tokenwait                                          |
|FODS ONLY: collisions:    garbledfrm:     garbledtrxm:        |
|  print_network_results  query_network_results  end           |
---------------------------------------------------------------
```

Figure 6.14: Network Wide Results Screen

```
-----------------------------------------------------------------
|            Query Network Wide Results                         |
|            simrunname:        protocol:                       |
|numrotation:   totofferedld:   throughput:   utilzation:       |
|msgdelv:       synfrmdelv:     asyn2frmdelv: asyn1frmdelv:     |
|frmdelv:       ttxr:           qf:                             |
|                   Average Maximum Minimum Std Deviation       |
|load to load delay                                             |
|              synch                                            |
|              asyn2                                            |
|              asyn1                                            |
|          all types                                            |
|link to link delay                                             |
|              synch                                            |
|              asyn2                                            |
|              asyn1                                            |
|          all types                                            |
|load to link delay                                             |
|              synch                                            |
|              asyn2                                            |
|              asyn1                                            |
|          all types                                            |
|link to load delay                                             |
|              synch                                            |
|              asyn2                                            |
|              asyn1                                            |
|          all types                                            |
|FDDI ONLY: tokenwait                                           |
|FODS ONLY: collisions:    garbledfrm:     garbledtrxm:         |
|  help  blankform  go  lastquery  exit                         |
-----------------------------------------------------------------
```

Figure 6.15: Network Wide Results Screen Query Mode

```
-----------------------------------------------------------------
|                Query Station Specific Results                   |
|simrunname:       protocol:          nodenumber:                 |
|ttxr:             qf:                numfrmq:                    |
|offeredld:        netbufavail:       orejectfrm:                 |
|trejectfrm:       msgdelv:                                       |
|sfrmxmit:    frmxmita1:     frmxmita2:      frmxmit:             |
|sdelvfrm:    delvfrma1:     delvfrma2:      delvfrm:             |
|                                                                 |
|                   Average Maximum Minimum Std Deviation         |
|queue length                                                     |
|received msg size                                                |
|delivered msg size                                               |
|load to load delay                                               |
|             synch                                               |
|             asyn2                                               |
|             asyn1                                               |
|         all types                                               |
|link to link delay                                               |
|             synch                                               |
|             asyn2                                               |
|             asyn1                                               |
|         all types                                               |
|load to link delay                                               |
|             synch                                               |
|             asyn2                                               |
|             asyn1                                               |
|         all types                                               |
|link to load delay                                               |
|             synch                                               |
|             asyn2                                               |
|             aync1                                               |
|         all types                                               |
|FDDI ONLY:tokenwait                                              |
|FODS ONLY:          garbledfrm:          garbledtxm:             |
| print_station_results  query_station_results  end               |
-----------------------------------------------------------------
```

Figure 6.16: Station Specific Results Screen

```
---------------------------------------------------------------
|                  Query Station Specific Results               |
|simrunname:       protocol:        nodenumber:                 |
|ttxr:             qf:              numfrmq:                    |
|offeredld:        netbufavail:     orejectfrm:                 |
|trejectfrm:       msgdelv:                                     |
|sfrmxmit:   frmxmita1:    frmxmita2:     frmxmit:              |
|sdelvfrm:   delvfrma1:    delvfrma2:     delvfrm:              |
|                                                               |
|                  Average Maximum Minimum Std Deviation        |
|queue length                                                   |
|received msg size                                              |
|delivered msg size                                             |
|load to load delay                                             |
|            synch                                              |
|            asyn2                                              |
|            asyn1                                              |
|        all types                                              |
|link to link delay                                             |
|            synch                                              |
|            asyn2                                              |
|            asyn1                                              |
|        all types                                              |
|load to link delay                                             |
|            synch                                              |
|            asyn2                                              |
|            asyn1                                              |
|        all types                                              |
|link to load delay                                             |
|            synch                                              |
|            asyn2                                              |
|            aync1                                              |
|        all types                                              |
|FDDI ONLY:tokenwait                                            |
|FODS ONLY:           garbledfrm:          garbledtxm:          |
| help  blankform  go lastquery  exit                           |
---------------------------------------------------------------
```

Figure 6.17: Station Specific Results Screen Query Mode

6.3 -APPENDIX C-

6.3.1 LANESBATCH.LOG

```
$ SET NOCONTROL_Y
$!
$ SET NOON
$ SET COMMAND SYS$SYSROOT:[SYSEXE]XMODEM.CLD
$ RUN SYS$SYSTEM:FIXNAME
$!
$! Modify system disk file protection default from
$! from W:RWED,O:RWED,G:RE,W to W:RWED,O:RWED,G:RE,W:RE
$!
$ SET PROTECTION=(G:RE,W:R) /DEFAULT
$ !
$ if F$MODE() .eqs. "NETWORK" then EXIT
$ sys$name :== "PLU"
$!
$ if "BATCH" .eqs. "BATCH" then goto BATCHON !Skip symbol definitions on
$ BATCHON:
$ if F$MODE() .eqs. "BATCH" then DEFINE SYS$PRINT NLAO:
$ set CONTROL=(Y,T)
$ if f$search("login.com").nes."" then @LOGIN
$define term_ingres "vt100"
$define ing_gbf_device "vt100"
$set protection=(system:rwed,Owner:rwed,Group:RWED,World:R)/default
$abf_dir :== "set def sys_ingres:[ingresabf.lanes3db.lanes3app]"
$lanes3  :== $plu1:[richardso.image]lanes3app.exe -dlanes3db main
$in      :==  lanes3db
$ab      :==  lanes3db
$eve     :== ed/tpu
$write sys$output "  "

$write sys$output "LANES 3 release account"
LANES 3 release account
$exit

The header information produced in batch mode by the LANES 3 simulation.
  LANES3       job terminated at  3-OCT-1986 10:46:06.91
Accounting information:
  Buffered I/O count:            44      Peak working set size:   834
  Direct I/O count:              35      Peak page file size:    1681
  Page faults:                 1778      Mounted volumes:            0
  Charged CPU time:     0 00:00:03.99    Elapsed time:      0 00:00:06.88
```

Figure 6.18: LANESBATCH.LOG

6.4 -APPENDIX D-

6.4.1 Using Scripts with LANES 3

LANES 3 provides an alternate method for entering load to the network model. The information is provided via a text file called SCRIPT.DAT. Using this method the characteristics of each message are individually specified. The Station parameters, message type and message size are used in scripts. The script also specifies source station, destination station, and time of arrival.

Scripts may be used in conjunction with messages from mathematical distributions, or may be the only source of load for a simulation. By default, scripts are not used.

The arrival of messages in mathematical distributions is completely unaffected by a script message. If a message arrives at a particular time as a part of a distribution, it still arrives at the same time and is of the same message type and size, whether a script is included or not. The overall effect of a script is to superimpose script traffic on the other traffic specified by mathematical distributions.

A message in the network model is completely specified with the 5 parameters in the script file. Those parameters which in INGRES screens are computed automatically when using distributions, need to be specified explicitly when using scripts. Mathematical distributions are not applied to any parameter of a script message, except absorption time. With absorption time script messages and distribution generated messages are treated the same.

The basic format of a script file is maintained in the file MODELSCRIPT.DAT. See page 57 for an example of a script file. Changes are made to this file using a text editor. When editing is complete, the final version of the file should be saved as SCRIPT.DAT. The file contains records with 5 variables displayed in columns. The columns are:

1. source station
2. destination station
3. message size
4. message type
5. time of arrival

Numbers in the file must be right aligned in in each column. Columns must not be shifted.

- Source station and destination station have the same meaning previously described in the network model (1 to number of stations, integer).
- Message size is in octets (integer).

- Message type 0 is synchronous, type 2 is high priority asynchronous, type 1 is low priority asynchronous.

- Time refers to the time of arrival, specified in microseconds with 2 digits after the decimal. This is called "absolute time", meaning the time scale is the same as the simulation time. Simulation time starts at 0 and ends with the user specified control parameter "runtime".

In the following example of a script, the source is station 2, the destination station is 4, message size is 900 octets, message type is synchronous, and time of arrival is at 150.00 microseconds.

```
2 4    900 0       150.00
```

Time of arrival may also be specified as "relative" instead of "absolute". The message arrival time is relative to the arrival time of the previous message in the script file.

In the following example the message defined in the first example is followed by a second message: source station 1, destination station 4, message size 900, message type low priority asynchronous. The negative sign preceding the time of arrival means that it is relative to the arrival of the previous message. The arrival is 150.00 + 1000.00 = 1150.00 microseconds.

```
2 4    900 0       150.00
1 4    900 1     -1000.00
```

Thus, if a script message arrived at 141234.34 simulation time and the following script message's arrival was 2000.00 ms. later, the second message could be specified as "-2000.00" rather than 143234.34. With this method a new message can be inserted into a series of relative messages without changing all of the messages individually. Time, when specified relatively, is in relation to the previous script message in SCRIPT.DAT. This is true regardless of the source station, destination station, message type and message size of the previous message. A relative time may be offset from another script message which was also relative.

- The order of messages in the script file is not important, except when times are specified as relative.

- The relation of script messages to distribution messages is not examined by the simulation, so script messages can be sent to stations which are not a part of a distribution message class, or arrive at exactly the same time as another message.

- Script messages which arrive after the end of the simulation are ignored.

- Script messages must not be specified to have the same source as destination.

- The delays and other statistical reporting for script messages is exactly the same as if the messages were generated by distributions. For example, the addition of script messages will always result in an increase in Total Offered Load.
- There is no limit to the number of script messages, beyond the regular limitations to the network model.

The file for scripts is named SCRIPT.DAT. If this file exists when a simulation is run the file will be read, and each of the messages specified in the file will be added. When a simulation is finished running, the SCRIPT.DAT file is renamed using the simulation name input by the user and placed in [.oldfiles] subdirectory along with the simulation's other input files.

The Star*Bus version of the SCRIPT.DAT file is renamed *SIMRUNNAME*.FSCRIPT. The FDDI version is renamed *SIMRUNNAME*.SCRIPT. If no SCRIPT.DAT file exists in the [lanes3] directory, then only the messages specified on the Station Parameters screen are used by the network model.

```
2 4    999 0      700.00
2 4    999 0     -300.00
2 4    999 0     -800.00
2 4    999 0     -200.00
2 4    999 0     -700.00
2 4    999 0       -1.00
2 4    999 0     -100.00
2 4    999 0     -500.00
2 4    999 0     -300.00
2 4    999 0     -800.00
2 4    999 0     -600.00
2 4    999 0     -100.00
2 4    999 0     -500.00
2 4    999 0     -800.00
2 4    999 0     -900.00
2 4    999 0     -400.00
2 4    999 0     -900.00
2 4    999 0     -100.00
2 4    999 0     -900.00
2 4    999 0     -600.00
2 4    999 0     -700.00
2 4    999 0     -300.00
2 4    999 0     -400.00
2 4    999 0     -900.00
2 4    999 0     -500.00
2 4    999 0     -600.00
2 4    999 0     -900.00
2 4    999 0     -200.00
2 4    999 0     -400.00
2 4    999 0     -900.00
```

Figure 6.19: Example of a SCRIPT.DAT file

6.5 -APPENDIX E-

6.5.1 Using Traffic Classes with LANES 3

LANES 3 provides a method of randomly assigning destinations among classes of stations, rather than among all stations in the simulation without discretion.

- Up to 10 classes can be specified.
- 1 or all stations can be within a class.
- Each station belongs to a unique class.
- All stations have a class; without using procedures described here all stations must be of class 1.
- Traffic can be sent from a class to other stations of its own class or to other classes, and from several classes to a single class.

The class of a station is entered on the Station Specific screens. The parser in the "add" command can be used to assign a class number to all stations in a class at one time. Traffic can be sent within a class - but in this case there must be more than one station in the class.

There are three traffic matrix files for each protocol. The Star*Bus (FODS) protocol files are named: FODS30.DAT, FODS31.DAT, FODS32.DAT. The FDDI protocol files are named: RING30.DAT, RING31.DAT, RING32.DAT. The portion of the filename "30" labels the file for synchronous messages, "31" for priority 1 asynchronous messages, "32" for priority 2 asynchronous messages. Apart from the names, all six files have the same format. See page 60 for a copy of one of the input files.

When the simulation runs under normal conditions this default version of the input files is used. The numeric rows represent sending classes, the columns the receiving classes. In the default version of the files only one traffic class exists, thus 100% of the traffic in class 1 is sent to class 1 stations.

The distribution among classes is normalized, so any values inside the matrix table between 0 and 100 are allowable along a row, and need not add to any particular number. For example, if there are to be 3 message classes then the input files need to be edited. The second attached example shows a possible set of new values. The first 10 characters of the line are for comments, it is a good idea to include the simulation run name here. In this example class 1 sends to itself, to class 2 and to class 3:

```
simrunname 1     2  1  1  0  0  0  0  0  0  0
```

It sends to other stations in its class twice as much (on the average) as to classes 2 and 3. All undefined classes naturally have "0" proportion of the traffic.

```
simrunname 2      1  0  0  0  0  0  0  0  0  0
```

Class 2 sends only to class 1.

```
simrunname 3      0  0  1  0  0  0  0  0  0  0
```

Class 3, let us say, only contains one station. Therefore this 3rd row contains a mistake, since a station cannot send messages to itself. This error would stop the simulation. A corrected row, where class 3 is sending to 2 only might be:

```
simrunname 3      0  1  0  0  0  0  0  0  0  0
```

It would be equally correct, and mean the same thing to enter for the 3rd row:

```
simrunname 3      0 100 0  0  0  0  0  0  0  0
```

which the user may find easier to read.

To change these files the user may use any text editor. It is important to preserve the format of the files, so care must be taken not to delete rows, and to line columns in the same place as in the default files. The left digit in the number lines up with the carets on the bottom row of the files. The program is intolerant of errors, and will abort.

When the simulation runs it reads only the latest version of these three files. When the simulation concludes the latest version of these files is copied into the [.oldfiles] subdirectory along with the simulation's other input and output files.

In the VMS operating system a file is specified by name, eg. FODS30.DAT and by version ";2". The VMS command "dir FODS3*.DAT" will produce a listing of all the Star*Bus input files which exist in the main directory. The highest version of the file will be used by the simulation. The file FODS30.DAT;1 is the first version of the file, it is also the default version. This file is protected and cannot be deleted by mistake. Any changes made to these files will not appear in the default version, but the computer will create a higher numbered version. That higher version will be used, only once, when the next simulation is run.

```
simrunname
simrunname   1    2   1   1   0   0   0   0   0   0   0
simrunname   2    1   0   0   0   0   0   0   0   0   0
simrunname   3    0   0   1   0   0   0   0   0   0   0
simrunname   4    0   0   0   0   0   0   0   0   0   0
simrunname   0    0   0   0   0   0   0   0   0   0   0
simrunname   0    0   0   0   0   0   0   0   0   0   0
simrunname   0    0   0   0   0   0   0   0   0   0   0
simrunname   0    0   0   0   0   0   0   0   0   0   0
simrunname   0    0   0   0   0   0   0   0   0   0   0
simrunname   0    0   0   0   0   0   0   0   0   0   0
```

traffic
class

Figure 6.20: Example of file RING30.DAT

6.6 -GLOSSARY-

-A-

absorption time
unit: microseconds
station specific input parameter
FDDI/Star*Bus
The period of time which a fully received message waits in the Network Layer as it is being absorbed by the Load Layer. Only one message may be in the process of absorption at a time.

type
Absorption time specified by one of four distribution types:
1 = Constant
2 = Exponential
3 = Normal (Gaussian)
4 = Uniform

var 1
First parameter for absorption time located on station specific input screen.
Set var 1 to Mean for types 1 - 3.
Set var 1 to lower bound for type 4 (Uniform).

var 2
Second parameter for absorption time located on station specific input screen.
Set to Standard Deviation for types 1 - 3.
Var 2 should be set to zero for type 1 (Constant) and greater than zero for type 2 (Exponential).
Var 2 should be set to upper bound for type 4 (Uniform).

acknowledgement
FDDI/Star*Bus
The acknowledgement or negative acknowledgement modeled in LANES 3 is used to indicate whether there was enough space in the Network Layer buffer at the destination station to receive the data.

armlength
unit: meters
station specific input parameter
Star*Bus
Distance from the center of the star network to the station.

-B-

backofftime
back off time
unit: microseconds
network wide input parameter
FDDI/Star*Bus
The backoff period commences when the transmitter receives a NACK. It lasts until backofftime expires, or until an ACK is received. Meanwhile frames for that destination are blocked at the Network Layer in the sending station.

bandwidth
bandwidth
unit: normalized to %
station specific input parameter
FDDI
Percent of the total synchronous allocation assigned to this station. (Synchronous allocation is the parameter "percentsyn" on the Network Parameter input screen.) Maximum total time allowed for synchronous transmissions at this station each time it receives the token. Values entered here are normalized among all stations. That is, whatever numbers are entered, the aggregate is adjusted so that it equals 100 percent of the synchronous bandwidth.

bufferdelay
buffer delay
unit: microseconds
network wide input parameter
Star*Bus
Time required to fill the Link Layer transmission buffer with a frame from the Network Layer.

-C-

class
unit: integer
station specific input parameter
FDDI/Star*Bus
Integer index of the traffic class to which this station belongs. For this implementation of the network model the value should be 1.

collisions
unit: integer
network wide output parameter
Star*Bus
The number of frame collisions that occurred. A single collision may include 2 or more frames. Appears in Network Wide Results report and query screen.

comments1 comments2 comments3
control input parameter
User defined comments, restricted to 57 characters per line.

-D-

database
All input and output information, essential for the running of the network model, is stored in a database. Data in the database may be retrieved, added, modified, or deleted using the LANES 3 network model screens.

delivered
Message (or frame) was succcessfully received by destination station.

delvfrm... (sdelvfrm, delvfrma2, delvfrma1)
delivered frames
unit: integer
station specific output parameter
FDDI/Star*Bus
Number of frames delivered for all types (synchronous, asynchronous priority 2 and 1). Appears in Station Specific Results report as Delivered Frames.

delvmsgsz (delvmsgszav, delvmsgszmax, delvmsgszmin)
delivered message size
unit: octets
network wide and Station Specific output parameters
FDDI/Star*Bus
Size of messages delivered by this station (average, maximum, minimum). Appears in Network Wide Results and Station Specific Results reports as Delivered Message Size.

-E-

elasticdelay
elasticity buffer delay
unit: bits
network wide input parameter
FDDI
Internal station delay at the Physical Layer. Elasticity delay is expressed in bit times, (the amount of time it takes to transmit one bit).

-F-

FODS
Fiber Optic Demostration Systems (FODS) star bus, based on a prototype hardware version of the protocol developed by the Sperry Corporation. Also referred to as Star*Bus in this document.

frame
Messages are divided into frames. The largest frame size parameter sets the upper bound on size. All data from Network Layer of transmitting station to Network Layer of receiving station is in frames.

... frmdelv (synfrmdelv, asyn2frmdelv, asyn1frmdelv, frmdelv)
frames delivered
network wide output parameter
FDDI/Star*Bus
Number of frames delivered for the entire network. Appears in Network Wide Results report as # Frames Delivered.

...frmxmit... (sfrmxmit, frmxmita2, frmxmita1)
frames transmitted
station specific output parameter
FDDI/Star*Bus
Number of frames transmitted for each station. Appears in Station Specific Results report as Frames Xmitted by message types (Synch, Asynch-2, Asynch-1, All Types).

-G-

gaptime
unit: microseconds
network wide input parameter
Star*Bus
Amount of time that a station must sense that the channel is clear before it may transmit.

garbledfrm
garbled frames
network wide and station specific output parameter
Star*Bus
Total number of distinct frames which were involved in collisions. A frame is counted once regardless of the number of collisions it was involved in. Appears in Network Wide Results and Station Specific Results reports as Number of Frames Garbled.

garbledtrxm
garbled transmission
network wide and station specific output parameter
Star*Bus
Total number of transmissions that became involved in collisions. This counter is incremented each time a frame is involved in a collision, so that multiple collisions of a single frame are counted. Appears in Network Wide Results and Station Specific Results reports as Number of Transmissions Garbled.

-H-

headersize
header size
unit: octets
network wide input parameter
FDDI/Star*Bus
Number of octets appended to each frame, to represent both the header and the trailer.

-I-

idle octets
see numidleoctet

INGRES
Relational database product that the LANES 3 model uses to run the user interface and store data.

interarrival time
unit: microseconds
station specific input parameter
FDDI/Star*Bus
Specifies distribution for the time between successive arrivals of the end of messages at this station. To have no messages arrive, enter a constant distribution with var 1 greater than run time. Decreasing the interarrival time will increase CPU usage on the VAX.

type
Interarrival time specifed by one of four distribution types:
1 = Constant
2 = Exponential
3 = Normal (Gaussian)
4 = Uniform

var 1
First parameter for interarrival time located on station specific input screen. For type 1 (Constant) var 1 is constant interval, type 2 (Exponential) var 1 is mean, type 3 (Normal) var 1 is mean, type 4 (Uniform) var 1 is lower bound.

var 2
Second parameter for interarrival time located on station specific input screen. Type 3 (Normal) var 2 is standard deviation, type 4 (Uniform) var 2 is upper bound.

-L-

largestframe
unit: octets
network wide input parameter
FDDI/Star*Bus
When a message is broken into frames at the Network Layer it is broken into uniform pieces of the largest frame size allowed. Any smaller messages and left over remnants of a long message are converted to frames smaller than the largest size.

link to link delay
unit: microseconds
network wide and station specific output parameter
FDDI/Star*Bus
Frame delivery delay (average, maximum, minimum, standard deviation) from time of first entry to Link Layer at source station to time of acceptance at Link Layer of destination station. This value includes retranmissions. Appears in Network Wide Results and Station Specific Results reports as LINK TO LINK FRAME DELIVERY DELAY.

link to load delay
unit: microseconds
network wide and station specific output parameter
FDDI/Star*Bus
Frame delay (average, maximum, minimum, standard deviation) at the destination station measured from the time it appears in the Link Layer to the time the corresponding message has begun to be absorbed by the Load Layer. Appears in Network Wide Results and Station Specific Results reports as LINK TO LOAD MESSAGE DELIVERY DELAY.

linkquesize
unit: number of frames
station specific input parameter
FDDI
Size of the Link Layer queue for transmitting frames. Receiving Link Layer queue is always zero regardless of the value for this parameter.

load to link delay
unit: microseconds
network wide and station specific output parameter
FDDI/Star*Bus
Frame delay (average, maximum, minimum, standard deviation) at source station, measured from the time the corresponding message is queued at the Load Layer to time the frame is first queued at the Link Layer. Appears in Network Wide Results and Station Specific Results reports as LOAD TO LINK FRAME QUEUEING DELAY.

load to load delay
unit: microseconds
network wide and station specific output parameter
FDDI/Star*Bus
Message delivery measured from the time the message is queued at the Load Layer of the transmitting station to the time the message begins absorption at the Load Layer of the the receiving station. Appears in Network Wide Results and Station Specific Results reports as LOAD TO LOAD MESSAGE DELIVERY DELAY.

-M-

maxtimestep
maximum time step
network wide input parameter
Star*Bus
Maximum number of random delay timesteps when network changes from controlled access mode to random access mode. See **rdmstepsize**. The number of timesteps in a particular random delay is drawn from a uniform distribution on the range of integers from 1 to the value of this parameter.

messages
FDDI/Star*Bus
The model uses three types of messages:
synchronous
asynchronous priority 2
asynchronous priority 1

msgdelv
messages delivered
network wide and station specific output parameter
FDDI/Star*Bus
Number of messages completely delivered by all stations. Appears in Network Wide Results report as MESSAGES DELIVERED.

... msgsz (synmsgsz, asyn2msgsz, asyn1msgsz)
message size
unit: octets
station specific input parameter
FDDI/Star*Bus
Size of Synchronous, Asynchronous 2, Asynchronous 1 type messages transmitted by this station.

size
Distribution of the size of the messages transmitted by this station.
Specified by one of the four distribution types:

1 = Constant
2 = Exponential
3 = Normal (Gaussian)
4 = Uniform

var 1
Expressed as the total number of octets in the information field of a message.
Set var 1 to Mean for types 1 - 3.
Set var 1 to lower bound for type 4 (Uniform).

var 2
Set var 2 to standard deviation for types 1 - 3.
Var 2 will default to 0 if distribution type is 1 (Constant).
Set var 2 to upper bound for type 4 (Uniform).

netbufavail
Network Layer buffer available
unit: bytes or octets
station specific output parameter
FDDI/Star*Bus
Amount of Network Layer receive buffer space currently available. Appears in Station Specific Results report as Network Layer Available.

netrecevsize
Network Layer receive buffer size
unit: octets
station specific input parameter
FDDI/Star*Bus
Capacity of the Network Layer receiving buffer. Must be greater than largest frame size.

netxmitsize
Network Layer transmit buffer size
unit: octets
station specific input parameter
FDDI/Star*Bus
Capacity of the Network Layer transmitting buffer. Must be greater than largest frame size.

nextdistance
next distance
unit: meters
station specific input parameter
FDDI
Distance from the current station to the downstream station.

nodenumber
node number
station specific input parameter
FDDI/Star*Bus
Integer index of the station to which the parameter values on this screen apply. Stations (nodenumber) start at 1.

numfrmq
number of frames queued
station specific output parameter
FDDI/Star*Bus
Number of frames which entered the Network Layer transmit buffer. Appears in Station Specific Results report as #Frames Entered.

numidleoctet
number of idle octets
unit: octets
network wide input parameter
FDDI
Number of idle octets which are sent by a station after it receives a token but before it begins sending frames.

numrotation
number of token rotations
FDDI
Number of completed cycles of the token around the ring.

numstations
number of stations
network wide input parameter
FDDI/Star*Bus
Number of stations (also called nodes) in the network to be modeled. Minimum number of stations is 2. Maximum number of stations for FDDI is 100. Maximum number of stations for Star*Bus is 32. Increasing the number of stations increases CPU usage on the VAX.
Stations are numbered consecutively from 1 to the value of this parameter. In FDDI the token moves by ascending station numbers.

-O-

offeredld
offered load
unit: megabits per second
station specific output parameter
FDDI/Star*Bus
The number of message data bits generated at the Load Layer of this particular station divided by the statistics collection period. Appears in Station Specific report as Offered Load.

orejectfrm
other rejected frames
station specific output parameter
FDDI/Star*Bus
Number of rejections by this station, due to full receive buffer, of frames transmitted from other stations. Appears in Station Specific Results report as #Frames Rejected By This Node.

-P-

percentsyn
percent of synchronous traffic
unit: % of TTRT
network wide input parameter
FDDI
Percent of the TTRT which is to be assigned for synchronous traffic. This percent is further allocated between stations by the bandwidth input parameter.

primarypur
control input parameter
Brief text comment on purpose of simulation run.

priorthresh
priority threshold time increment
unit: microseconds
network wide input parameter
FDDI
Transmission of asynchronous messages is governed by the amount of time remaining on the THT (Token Holding Timer). In deciding whether or not there is sufficient time on the THT to allow transmission of an asynchronous frame, a station multiplies priorthresh by the frame priority, and compares the product with the value on the THT. If the value is greater than or equal to the THT value, then that frame is eligible for transmission.

-Q-

qf
queued frames
station specific output parameter
FDDI/Star*Bus
Total number of frames currently queued (at end of run) in the Network Layer transmit buffers of all the stations. Appears in Station Specific Results report as Queue Length.

quelen ... (quelngthav, quelenmax, quelenmin)
queue length
unit: number of frames
station specific output parameter
FDDI/Star*Bus
Number of frames (average, maximum, minimum) in the Network Layer transmit buffer. Appears in Station Specific Results report as Network Layer Transmit Buffer Queue Length.

-R-

rcdmsgsz ... (rcdmsgszav, rcdmsgszmax, rcdmsgszmin)
received message size
unit: octets
station specific output parameter
FDDI/Star*Bus
Size of messages (average, maximum, minimum) received by this station. Appears in Station Specific Results report as Received Message Size.

rdmstepsize
random delay stepsize
unit: microseconds
network wide input parameter
Star*Bus
Duration of each timestep in the random delay when network changes from controlled-access mode to random-access mode.

runtime
run time
unit: microseconds
control input parameter
Amount of simulated run time. This value should not exceed 350,000 microseconds.

-S-

signalspeed
signal speed
unit: meters per second
network wide input parameter
FDDI/Star*Bus
Propagation speed of the signal along the fiber optic cable.

simrundate
simulation run date
control input parameter
Date on which the simulation is run in YY/MM/DD format. Example: 86/01/09 = January 09, 1986. Simulation runs are retrieved on the History screen with the most recent "simrundate" at the top.

simrunname
simulation run name
Main Menu input parameter
Name given to this simulation which, along with the simulation's protocol, uniquely identifies the simulation in the database. It may include any combination of 10 letters and numbers (first character must be a letter).

startstatcol
start statistics collection
unit: microseconds
control input parameter
Point in simulated time at which collection of output statistics is to begin. If you expect start-up transient behavior which is atypical of long-run network performance, it is useful to set this parameter at a value exceeding the duration of the transient period. Statistics are not collected before this point. Frames are numbered from simulation time 0 (zero) regardless of the value of the startstatcol parameter.
starttrace
start trace
unit: microseconds
control input parameter

Point in time at which an event time trace is to begin. Trace output is only produced for a maximum of 2000 microseconds after this point. Trace output is sent to the *SIMRUNNAME*.TRACE text file in the VMS subdirectory [lanes3.oldfiles].

subperiods
unit: microseconds
control input parameter
Duration of interval(s) over which output statistics are to be accumulated. If set to zero or to a value exceeding the difference between the runtime and the time to start statistics collection, there will only be one such interval; otherwise, statistics are reported and cleared at the end of each such interval and a new interval immediately starts. This feature cannot be used in runs whose results you wish to store in the database.

-T-

throughput
effective throughput
unit: percentage
network wide output parameter
FDDI/Star*Bus
Number of bits delivered (excluding headers) as a percent of the number of bits that could have been delivered during the same time period. Retransmissions are ignored when computing throughput. Appears in Network Wide Results report as Effective Throughput.

et = effective throughput
nbd = number of bits delivered
tr = transmission rate
scp = statistics collection period

$$et = \frac{nbd}{tr * scp} * 100$$

timesltwidth
time slot width
unit: microseconds
network wide input parameter
Star*Bus
Duration of the timeslots in the controlled-access mode of the Link Layer protocol.

toknwait... (toknwaitav, toknwaitmax, toknwaitmin, toknwaitsd)
token wait
unit: microseconds
network wide and station specific output parameter
FDDI
Time spent waiting, (average, maximum, minimum, standard deviation), with data ready, for next usable token. This delay is associated only with the frame at the beginning of the queue. Appears in Network Wide Results and Station Specific Results reports as Wait Time For Usable Token.

totofferedld
total offered load
unit: megabits per second
network wide output parameter
FDDI/Star*Bus
The number of message data bits generated for the entire network at Load Layer divided by statistics collection period. Appears in Network Wide Results report as Total Offered Load.

to = total offered load
nbl = number of bits loaded
scp = statistics collection period

$$to = \frac{nbl}{scp}$$

transmission rate
see xmissionrat

trejectfrm
this station's rejected frames
station specific output parameter
FDDI/Star*Bus
Number of frames transmitted by this station that were rejected by other stations. Appears in Station Specific Results report as #Frames Rejected From This Node.

TTRT
target token rotation time
unit: microseconds
network wide input parameter
FDDI
Specifies the expected token rotation time.

ttxr
total time retransmitting rejects
unit: microseconds
network wide and station specific output parameter
FDDI/Star*Bus
Total time spent retransmitting frames rejected due to full destination receive buffer. Appears in Network Wide Results report as Time to Rexmit Rejects, and in the Station Specific Results report as Time Retransmit Rejects.

-U-

utilzation
utilization
unit: percentage
network wide output parameter
FDDI/Star*Bus
Number of bits transmitted as a percentage of total number of bits that could have been transmitted over the same time period. Includes transmission of headers and retransmission of rejected frames. Appears in Network Wide Results report as %Utilization.

u = utilization
nbt = number of bits transmitted
tr = transmission rate
scp = statistics collection period

$$u = \frac{nbt}{tr * scp} * 100$$

-X-

xmissionrat
transmission rate
unit: megabits per second
network wide input parameter
FDDI/Star*Bus
Transmission rate of the stations.